Durham University

Durham Research Online

Deposited in DRO:

07 April 2020

Version of attached file:

Accepted Version

Peer-review status of attached file:

Peer-reviewed

Citation for published item:

Wang, Yinduo and Zhang, Haofeng and Wang, Shidong and Long, Yang and Yang, Longzhi (2020) 'Semantic combined network for zero-shot scene parsing .', IET image processing., 14 (4). 757 -765.

Further information on publisher's website:

https://doi.org/10.1049/iet-ipr.2019.0870

Publisher's copyright statement:

This paper is a postprint of a paper submitted to and accepted for publication in IET image processing and is subject to Institution of Engineering and Technology Copyright. The copy of record is available at the IET Digital Library.

Additional information:

Use policy

The full-text may be used and/or reproduced, and given to third parties in any format or medium, without prior permission or charge, for personal research or study, educational, or not-for-profit purposes provided that:

- a full bibliographic reference is made to the original source
- a link is made to the metadata record in DRO
- the full-text is not changed in any way

The full-text must not be sold in any format or medium without the formal permission of the copyright holders.

Please consult the full DRO policy for further details.

Durham University Library, Stockton Road, Durham DH1 3LY, United Kingdom
Tel : +44 (0)191 334 3042 | Fax : +44 (0)191 334 2971
https://dro.dur.ac.uk

IET Journals
The Institution of Engineering and Technology

Semantic Combined Network for Zero Shot Scene Parsing

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

Yinduo Wang[1], *Haofeng Zhang*[1,✉], *Shidong Wang*[2], *Yang Long*[3], *Longzhi Yang*[4]

[1] *School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China.*
[2] *School of Computing Sciences, University of East Anglia, Norwich, UK.*
[3] *Department of Computer Science, Durham University, Durham, UK.*
[4] *Department of Computer and Information Science, Northumbria University, Newcastle upon Tyne, UK.*
✉*zhanghf@njust.edu.cn*

Abstract: Recently, image-based scene parsing has attracted increasing attention due to its wide application. However, conventional models can only be valid on images with the same domain of the training set, and are typically trained using discrete and meaningless labels. Inspired by the traditional zero shot learning methods which employ an auxiliary side information to bridge the source and target domains, we propose a novel framework called Semantic Combined Network (SCN), which aims at learning a scene parsing model only from the images of the seen classes while targeting on the unseen ones. In addition, with the assist of semantic embeddings of classes, our SCN can further improve the performances of traditional fully supervised scene parsing methods. Extensive experiments are conducted on the dataset Cityscapes, and the results show that our SCN can perform well on both Zero Shot Scene Parsing (ZSSP) and Generalized ZSSP (GZSSP) settings based on several state-of-the-art scene parsing architectures. Furthermore, we test our model under the traditional fully supervised setting and the results show that our SCN can also significantly improve the performances of the original network models.

1 Introduction

In the last few decades, image-based scene analysis has become one of the fundamental tasks of computer vision. It can be used in a large number of applications such as image editing and autonomous navigation, and attracts an increasing number of researchers to endeavor in it. The purpose of this task is to classify each pixel in the image by assigning discrete tag to it. Near recently, due to the rapid development of deep Convolutional Neural Networks (CNNs) [1, 2], many efforts have been devoted to employ CNNs to improve the performance of scene parsing, such as Fully Convolutional Network (FCN) [3], Pyramid Scene Parsing Network (PSPNet) [4] and Semantic Image Segmentation with Deep Convolutional Nets (DeepLab) [5], which can significantly outperform traditional feature extraction methods such as Conditional Random Field (CRF) [6].

However, in the era of big data, an increasing number of new scene categories are emerging. The traditional full supervision scene analysis methods are suffering from a serious problem, *i.e.*, when a new category of image appears, the model should be retrained by adding the samples of the new class. For example, if we have trained a fully supervised model with the images captured from a town scene, when the model is applied in a city scene, there will be many unseen categories such as *'skyscraper'*, which makes the model inoperable. Therefore, we need to acquire a large number of images from the new categories and combine them into the original data set to train a new model, which is a labor cost and difficult to be implemented.

Conventional Zero Shot Learning (ZSL) [7–10] is inspired by the behavior of our human beings that when we meet new categories, we always employ some intermediate information to build up a bridge from seen to unseen categories. Therefore, some semantic vectors such as attribute annotated by experts [11], have been utilized as the auxiliary information to achieve the purpose of recognizing novel categories. As shown in Fig.1(b), following the same strategy of conventional ZSL, Zero Shot Scene Segmentation (ZSSS) is first proposed by [12], which aims to train models only dependent on the annotated data within seen classes, but can be applied on a disjoint dataset of unseen classes. In addition, inspired by conventional Generalized Zero Shot Learning, as shown in Fig.1(c), in this paper we introduce a more realistic task Generalized ZSSP (GZSSP), which enlarges the search spaces into both seen and unseen classes.

Moreover, most of the state-of-the-art scene parsing networks adopt the same strategy that the labels of all pixels are discrete and barely have relations to each other. The labels used during training are always integers, *e.g.*, the category *'person'* is annotated as 1, and *'rider'* is 2. With such operation, it can be easily found that the difference between the label *'person'* and the label *'rider'* is same as that between *'person'* and *'building'*. However, we all know that the difference between *'person'* and *'rider'* is much smaller than that between *'person'* and *'building'* in realistic scenarios. Therefore, we argue that the labels are related, which should be adopted in scene parsing networks.

In order to solve the aforementioned problem, we propose a novel deep architecture called Semantic Combined Network (SCN), which combines the word embedding of classes into classification networks. Specifically, we no longer consider labels as meaningless values, but instead exploit a higher-dimensional attribute vector for each label, such as Word2Vec [13], which allows us to be able to work on both ZSSP and GZSSP settings at the same time. Therefore, employing the relationship between classes is much better to improve the performance of the corresponding original network. In this paper, based on several state-of-the-art architectures such as FCN, PSPNet and DeepLab, we utilize the proposed SCN to solve both ZSSP and GZSSP tasks. During the experiments, we randomly split the classes of Cityscapes[14] into two disjoint parts, including seen classes and unseen classes, and conduct extensive experiments on them. Besides, due to the combination of semantic information, we also test our SCN under the traditional fully supervised setting. The contributions of this work are as follows,

- We propose a novel and effective Semantic Combined Network (SCN), which integrates the labels' semantic embeddings into scene parsing network to solve the more challenging task, Zero-Shot Scene Parsing (ZSSP);
- Inspired by GZSL, we introduce a more realistic task, Generalized Zero-Shot Scene Parsing (GZSSP), which enlarges the search space from seen classes to all classes;

© The Institution of Engineering and Technology 2015

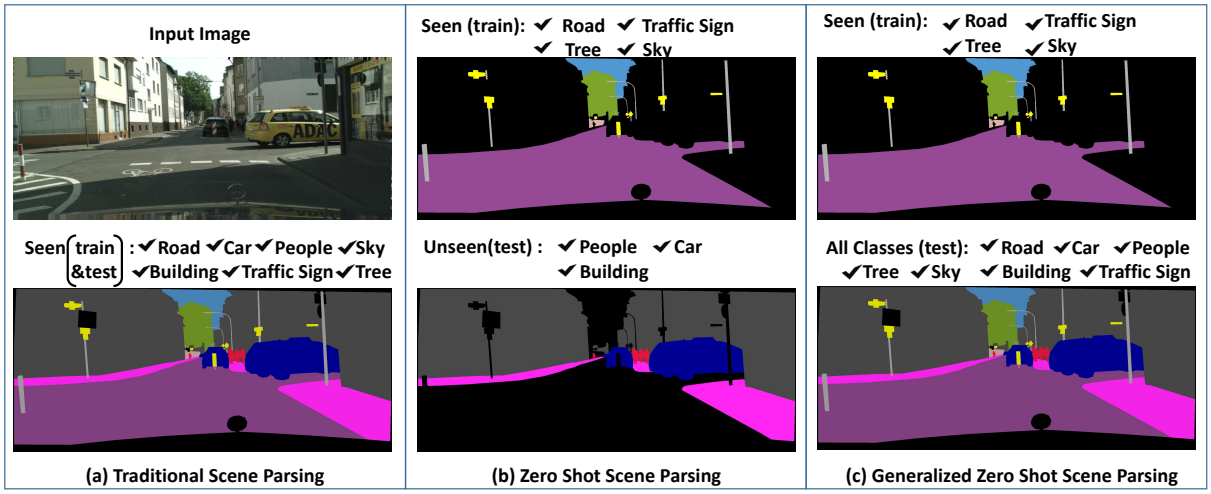

Fig. 1: An illustration of task ZSSP and our introduced Generalized ZSSP (GZSSP).

• Instead of directly employing meaningless and unrelated labels, we combine the semantic information of each class into several state-of-the-art networks, and make them feasible for new ZSSP and GZSSP tasks;
• Extensive experiments are conducted on all the three settings, including ZSSP, GZSSP and conventional fully supervised learning, and the experimental results show that our SCN can not only well solve the tasks of ZSSP and GZSSP, but also significantly improve the performance of the state-of-the-art methods under the conventional supervised setting.

The main content of this paper is organized as follows: In section 2 we briefly introduce the existing methods for Scene Parsing and ZSL. Section 3 describes the proposed method in detail. Section 4 gives the experimental results of comparison with existing methods for ZSSP, GZSSP, and traditional scene parsing. Finally, we conclude this paper in section 5.

2 Related work

2.1 Scene parsing and Semantic segmentation

Scene parsing, based on semantic pixel-wise segmentation, is an ongoing hot topic research [15–17]. Conventional methods often utilize contextual information of raw data or hand-craft feature for scene parsing, such as Conditional Random Field (CRF) [6], which combines multi-scale component to refine the performance. Based on CRF, Lucchi et al. [18] tried to make full use of global image-level prior to scene parsing. Based on some early efforts such as VGG [19], GoogLeNet [20], AlexNet on object classification with CNNs, FCN [3] replaces the last fully-connected layer of them with series of deconvolution layers to classify each pixel, and achieves great success. After realizing CNNs have such great potential and effect in this field, a large number of researchers have been endeavoring in it. Some works such as DilateNet [21] and DeepLab [5] are dedicated to enlarging the receptive field of convolutional layers to improve the performance, and PSPNet [4] has made a big progress by integrating the contexts of different scales. To refine the results, some methods such as Deeplab [5] and CRFasRNN [22] enhance the post-processing of the networks to improve the segmentation accuracy.

2.2 Zero Shot Learning (ZSL)

Zero-shot learning can be simply classified into Inductive ZSL and Transductive ZSL. Inductive ZSL methods are only trained with labeled seen data and unlabeled unseen data is strictly inaccessible, while transductive ZSL, which is first proposed by Y. Fu *et al.*[23], learns a model with both labeled seen and unlabeled unseen data. For inductive ZSL, since the visual attribute learning proposed by [11], many researchers conduct their work on establishing a visual to attribute projection to find the relationship between seen and unseen classes. Early efforts like DAP [24] learns probabilistic attribute classifiers to predict the label. ALE [25], SJE [26], and DEVISE[27] project features into a semantic embedding space by employing bilinear compatibility functions. CONSE [28] and Semantic Similarity Embedding (SSE) [29] exploit seen classes to improve the performance and reduce the usage of manual attributes by constructing the attributes of unseen classes. Near recently, Long *et al.* [30, 31] propose a generative framework that constructs a projection from attributes of seen classes to visual features, and then utilize the attributes of unseen classes to synthesize unseen visual features, finally a supervised model is trained for all classes. For transductive ZSL, semi-supervised framework [32] learns a multi-class classification model on all classes jointly with both labeled data and unlabeled data as input. In [33] Song *et al.* designed two independent objective functions for seen data and unseen data respectively, and integrates them into a whole framework during training phase so as to employ the powerful feature extraction ability of deep architecture.

2.3 Generalized Zero Shot Learning (GZSL)

ZSL assumes that the ascription of test data is known in advance, thus search space of the nearest neighbour can be restricted on only unseen classes. Chao *et al.* [34] firstly argued that under more practical situations, the convention ZSL is unreasonable, because we cannot obtain the knowledge whether the test data belongs to unseen classes or seen classes beforehand in most circumstances. Therefore, they propose the new task—Generalised ZSL, which assumes that the search space of the nearest neighbour should be extended to both seen and unseen classes. Recently, Zhang *et al.* in [35] considered GZSL problem as a triple verification problem and a novel optimization of regression and compatibility function is proposed to solve this problem. Subsequently, Xian *et al.* [36] put forward a new standard split of several popular datasets for GZSL testing, and release a benchmark of some recent ZSL methods, which makes the later researchers more convenient and has greatly promoted the development of ZSL.

2.4 Semantic Embeddings

Current ZSL methods always rely on the intermediate attributes, which represent the semantic embeddings of both seen and unseen classes. Conventional attributes [37, 38] are usually annotated by experts with real values. This type of annotation needs experts' prior

© The Institution of Engineering and Technology 2015

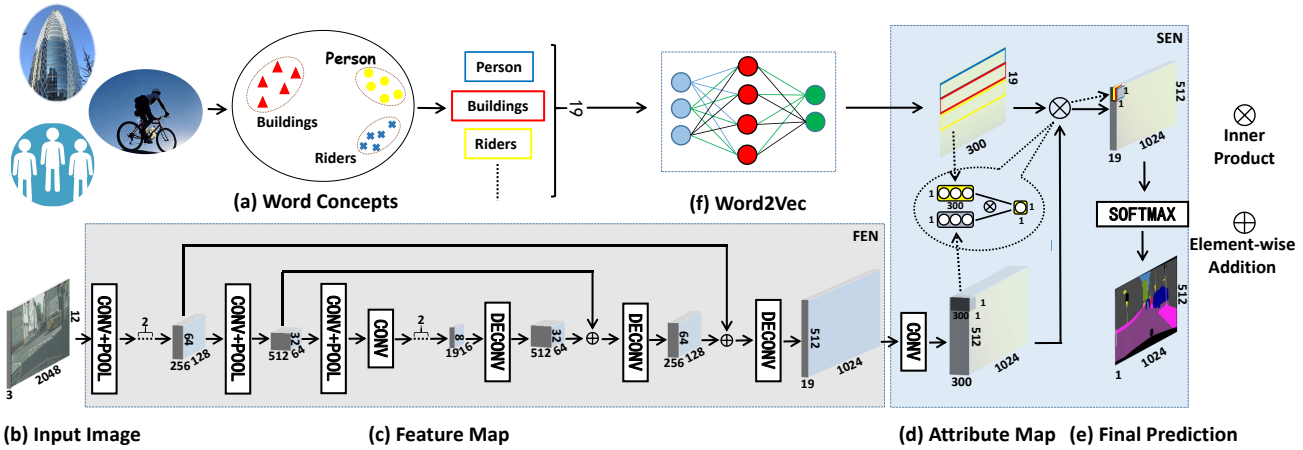

Fig. 2: An illustration of the framework of our proposed network. Besides the traditional CNN based scene parsing architectures, we add another branch to extract semantic information of labels to meet the ZSSP and GZSSP task. In the Word2Vec part, we employ an open-source pre-trained word embedding model to directly extract the word vectors for each class as the input to the following computation. During attribute map construction, the inner product of a pixel's visual attribute and each class's semantic attribute are calculated to generate the similarity of this pixel to all classes.

knowledge, and will cost a lot of manpower. Therefore, recent methods such as Attributes2Classname [39] start to use Word2Vec [13] to automatically generate attributes based on the dataset *'WikiPedia'*. However, this type of textual description often leads to serious performance degradation because it is not directly related to the visual appearance. Another semantic attribute representation is based on similarity, which can be annotated by humans [40, 41] or the textual descriptions [29, 42].

3 Methodology

In this section, we will firstly give the definitions of our tasks on three different settings, including ZSSP, GZSSP and fully supervised learning, and then describe the proposed SCN in details.

3.1 Problem Definitions

3.1.1 Preliminaries: Given a set of images X, including training part $X^s \in \mathbb{R}^{H\times W\times N_s}$ and testing part $X^r \in \mathbb{R}^{H\times W\times N_r}$, and all the classes of the images are denoted as $D = S \cup U$, where $S = \{1, ..., s\}$ represents the set of seen classes, and $U = \{s+1, ..., s+u\}$ stands for the set of unseen classes. To fulfill the requirement of the tasks, we divide all the image pixels into four parts: training pixels of seen classes in X^s are represented by X^s_s and test pixels of seen classes in X^r are denoted as X^r_s. Similarly, the training pixels in unseen classes are denoted as X^s_u and the test pixels of unseen classes are represented as X^r_u. Now, we can formally describe the scene parsing tasks with the following three types:

3.1.2 Zero Shot Scene Parsing (ZSSP): In ZSSP task, the given data only includes the pixels X^s_s of seen classes in X^s and their corresponding labels S during training, we need to train a function $F_z : X^s_s \rightarrow S$, which can be transferred to unseen classes to predict the labels of X^r_u. It should be noted that the search space is restricted on unseen classes only.

3.1.3 Generalized ZSSP (GZSSP): Similar as ZSSP, only X^s_s is given for GZSSP task in training phase to learn a function $F_g : X^s_s \rightarrow D$ to predict the labels of both X^r_s and X^r_u. In addition, different from ZSSP, we are supposed to be ignorant of whether the test pixel belongs to seen classes or unseen classes, so the search scope of this task should be conducted on both seen and unseen classes.

3.1.4 Traditional Fully supervised scene parsing: Given the pixels $X^s = X^s_s \cup X^s_u$ of all classes and their corresponding labels D, the purpose of this task is to train an optimal projection model $F_f : X^s \rightarrow D$ to assign each pixel with a label, and then the trained model can be applied on test data X^r.

3.2 Model Architecture

Since our SCN can be compatible with many architectures such as FCN, for the sake of simplicity of description, we here first take FCN as an example, and the extensions on PSPNet and DeepLab will be described later. The whole architecture of our proposed SCN is illustrated in Fig.2. Since SCN adopts the semantics of labels, we divide our framework into two parts, one is Feature Extraction Network (FEN) to extract the high dimensional feature from input images, another is Semantic Embedded Network (SEN), which is exploited to compute the semantic embeddings from labels and generate the category probability.

3.2.1 FEN: In our framework, as shown in the gray block of Fig.2, an image x is firstly input into the Feature Extraction Network (FEN) to generate a high dimensional features for each pixel. Here, the FEN can be replaced by many other scene parsing networks, such as FCN, PSPNet and DeepLab, by removing the final softmax layer.

In FEN, the input images $x \in X^s$ first pass through a VGG module and generate small scale feature, and then several deconvolutional blocks are added to scale the feature map to the same size as the input image, where the channel number equals the number of categories. Additionally, in order to obtain image information with different resolutions, some skip connections are added in the FEN, which can be seen in the gray block of Fig. 2. We first de-convolute the last layer of VGG into the same size of *'conv4'*, and then add the result and *'conv4'* in element-wise to generate *'deconv4'*, which is subsequently de-convoluted into the same size of *'conv3'*, and added by *'conv3'* in element-wise to get *'deconv3'*. Subsequently, the *'deconv3'* is up-sampled and convoluted to get the feature map with the same size as the original input image. For the convenience of being replaced by FCN or other architectures, we add an extra convolutional layer to generate the final coarse-category prediction map with the same channels as the number of categories. The final prediction map is denoted as $\mathcal{F}$.

3.2.2 SEN: SEN, the critical part of our network, is designed to extract the deep discriminative semantic representation information of labels to assist the classification of our method. The blue block in Fig.2 is the framework of our SEN, which can be divided into two parts, one is utilized to extract semantic information of labels and another is for generating the image semantics in our network.

© The Institution of Engineering and Technology 2015

In the first part, once we have obtained the ground-truth of the images, we will know the exact name of each category. Through the procedure of Word2Vec, which is a network pre-trained from large text corpora, we can obtain a M-dimensional vector by taking a label name as input. Therefore, for every single class $k \in D$, we can generate a M-dimensional vector v_k by Word2Vec, and finally achieve an $M \times (s+u)$ dimensional semantic matrix $\mathcal{V}_D$ for all classes.

In the second part, in order to make the coarse-category map to be consistent with dimension of the extracted semantic vectors, we convolute the coarse-category prediction map $\mathcal{F}$, which is the final output of FEN with $s+u$ channels, with a kernel ω to generate a high dimensional semantic map $\mathcal{S}_D$, where, the dimensionality of ω is $(s+u) \times 3 \times 3 \times M$, and $\mathcal{S}_D$ has its dimensionality of $H \times W \times M$.

So far, each pixel in image has an M-dimensional vector to represent its deep semantic information. Thus, the probability of each pixel x_{ij} belongs to a certain category can be calculated by applying an Inner Product and Softmax, which can be represented as,

$$p_{ijk} = \frac{e^{\mathcal{P}_{ijk}}}{\sum_{c \in D} e^{\mathcal{P}_{ijc}}}, \tag{1}$$

where, p_{ijk} is the probability of the pixel x_{ij} belongs to the k^{th} class. $\mathcal{P}_{ijk}$ is the $\{i,j,k\}^{th}$ entry of the matrix $\mathcal{P} = \mathcal{S}_D \mathcal{V}_D$.

3.3 Settings

3.3.1 ZSSP Setting: It is noteworthy that the pixel of unseen classes should not be included during ZSSP training. Since X_s^s and X_u^s might be found in the same image, we construct a mask to block the pixels belong to unseen classes when calculation the cross entropy loss to fulfill this task. We only choose the part of seen classes of semantic matrix $\mathcal{S}$ for train, and denote it as $\mathcal{S}_S \in \mathbb{R}^{M \times s}$, which is subsequently multiplied with the semantic embeddings $\mathcal{V}_S$ of seen classes. We use the same formula as Eq. 1 to calculate the probability of each pixel by replacing $c \in D$ and $\mathcal{P} = \mathcal{S}_D \mathcal{V}_D$ with $c \in S$ and $\mathcal{P} = \mathcal{S}_S \mathcal{V}_S$ respectively. And the cross-entropy loss function can be defined for ZSSP task as follows,

$$\begin{aligned} &\mathcal{L}_z(X_s^s, Y_s^s) \\ &= -\sum_{x \in X_s^s} \sum_{i,j \in x \& c \in S} (y_{ijc} log p_{ijc} + (1 - y_{ijc}) log(1 - p_{ijc})), \end{aligned} \tag{2}$$

where, Y_s^s is the corresponding label of X_s^s.

During testing, we extract unseen classes semantics, and denote them as a matrix $\mathcal{V}_U \in \mathbb{R}^{M \times u}$. Given a test pixel $\hat{x}_{ij}$ in an image x, we take x to pass through our SCN, where the word semantics uses $\mathcal{V}_U$, and obtain the final probability p_{ijc} of the pixel $\hat{x}_{ij}$ for all unseen classes U. The prediction of ZSL is calculated as,

$$\hat{y_{ij}} = \underset{c \in U}{argmax}\, p_{ijc}. \tag{3}$$

For the seen part of test images X_s^r, we can easily predict their labels with the same process of fully supervised task.

3.3.2 GZSSP Setting: In ZSSP setting, we previously assume that the test images can be divided into seen part and unseen part, thus for the unseen part, we just need to find the nearest neighbour only on unseen classes. However, ZSSP is usually unreasonable in realistic scenarios because we cannot know whether the pixel belongs to seen or unseen classes in advance. Therefore, generalizing the classification on all classes is necessary, and we call this task GZSSP. In this task, we calculate three metrics, test seen accuracy acc_{tr}, test unseen accuracy acc_{ts} and harmonic accuracy H. acc_{tr} and acc_{ts} represents class mean average precision from test seen data and test unseen data respectively, and H is defined as,

$$H = \frac{2 \times acc_{tr} \times acc_{ts}}{acc_{tr} + acc_{ts}}. \tag{4}$$

3.3.3 Fully Supervised Scene Parsing Setting: For the traditional fully supervised scene parsing setting, the sample pixels of both seen classes and unseen classes can be obtained during training, and we also train the whole framework using cross-entropy loss. The loss function can be represented as follow,

$$\begin{aligned} &\mathcal{L}_f(X^s, Y^s) \\ &= -\sum_{x \in X^s} \sum_{i,j \in x \& c \in D} (y_{ijc} log p_{ijc} + (1 - y_{ijc}) log(1 - p_{ijc})), \end{aligned} \tag{5}$$

where, $Y^s$is the corresponding label of X^s, and y_{ijc} is c^{th} entry of the one-hot vector label of pixel x_{ij}.

During testing, we predict the label of a pixel $\hat{x}_{ij} \in X^r$ by finding its nearest class based on the probability of different class embeddings, which can be easily generated from the last layer of our SCN with an argmax method,

$$\hat{y_{ij}} = \underset{c \in D}{argmax}\, p_{ijc}. \tag{6}$$

3.4 Other types of FEN

Since our SCN is an extended architecture based on conventional deep scene parsing models by adopting the word semantic embeddings, thus it is convenient to substitute the FEN with other scene parsing frameworks such as PSPNet and DeepLab, and the replaced models can be found in Fig. 3, where SE represents Semantic Embedded. For the SCN with PSPNet architecture, the upper branch in Fig. 3, we calculate the final model loss after the inner product of feature semantics and word embeddings of classes. And the lower branch in Fig. 3 is SE-Deeplab, instead of fusing all four atrous convolution layers directly to generate a prediction map, we project them to semantic space and then fuse the output results, which is multiplied by the word embeddings afterwards to yield the class probability of each pixel.

4 Experiments

4.1 Dataset

We employ the dataset Cityscapes[14] in our experiments. Cityscapes is a semantic urban scene parsing dataset, and it is released recently for pixel-wise scene parsing and instance annotation. The dataset is comprised of a large, veritable set of stereo video sequences, among which there are 5,000 images annotated with pixel-level high quality annotations. In the dataset, there are 19 predefined categories containing both stuff and objects. In our experiment, 3,475 images are assigned as train set and the left 1525 images are treated as test set. For the tasks of ZSSP and GZSSP, we randomly pick 5 classes as unseen classes and the remaining 14 as seen classes, and the results recorded in Tab.1 and Tab.2 are the average values of 10 executions.

4.2 Evaluation Metrics

For evaluation, there are four kinds of metrics used in our scene parsing experiments. For evaluating the precision of the tasks ZSSP and GZSSP, mean Average Precision (mAP), pixel-wise accuracy (Pixel Acc.), and H, which is introduced above, are employed in our experiments. Besides, for traditional fully supervised scene parsing task, we utilize the mean of class-wise intersection over union (mIoU).

4.3 Implementation Details

Since details are always important for training a neural network, we describe the settings of our SCN in this paragraph. Besides, for the convenience of description and comparison, we name our frameworks as SE-FCN (SE for Semantic Embedded), SE-PSP and SE-Deeplab respectively.

© The Institution of Engineering and Technology 2015

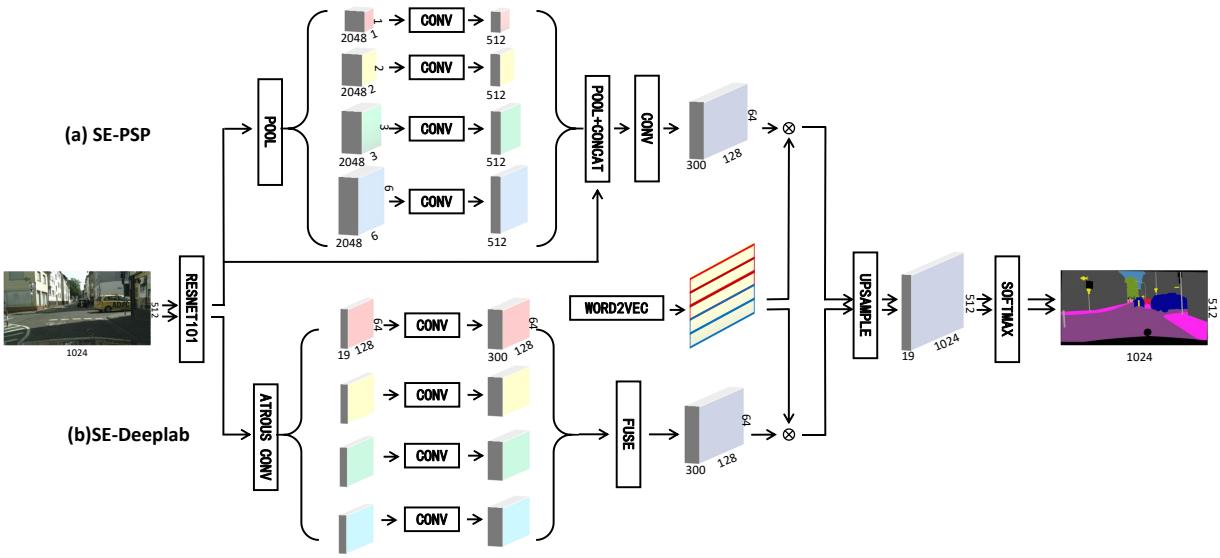

Fig. 3: An illustration of task ZSSP and our introduced Generalized ZSSP (GZSSP).

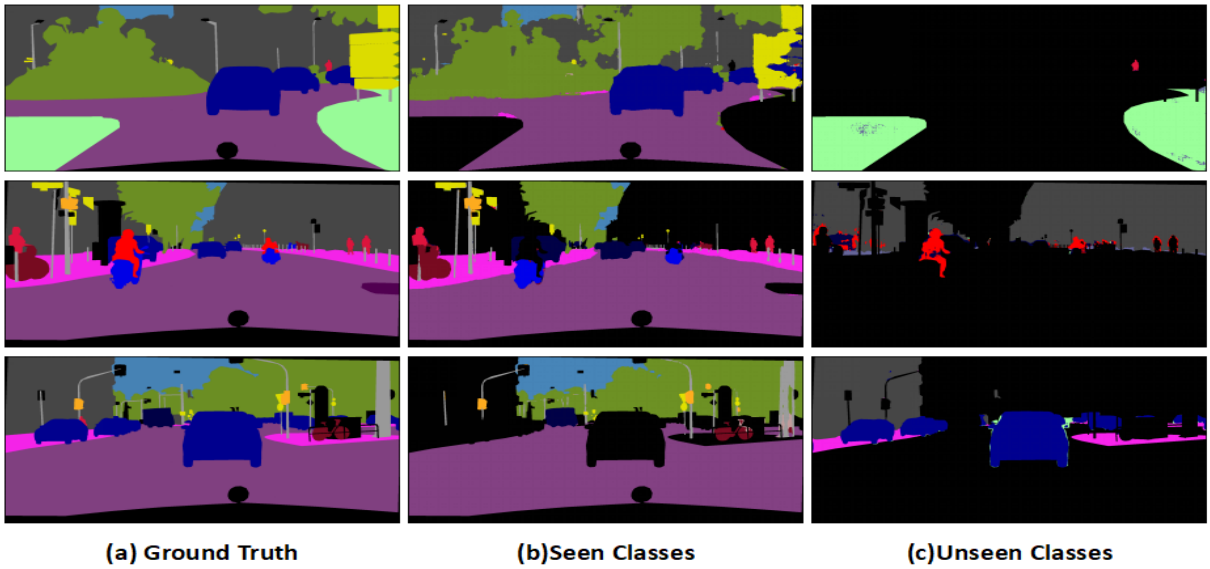

Fig. 4: Visual results of Zero Shot Scene Parsing. The rows from top to bottom are the prediction with SE-FCN, SE-DeepLab, and SE-PSPNet respectively.

We train our models with SEN fixed and FEN fine-tuned. Noted that during the experiment of SE-FCN, because the size of the original image is 1024×2048, we need $1024 \times 2048 \times 300 \times 4$ bytes to store the generated matrix of feature semantics, which is too big for a single GPU such as GTX 1080Ti. Therefore, We resize the original input images to $1/2$ of their original height and width by bi-linear interpolation, which might leads to performance degradation for information loss during interpolation, but we still observe a significant performance improvement comparing to their original methods, which can prove the effectiveness of our architecture.

In addition, we use a constant learning rate, which is set to 1×10^{-4}, and the iteration time is set to $10K$ for SE-FCN and $6K$ for SE-PSP and SE-Deeplab. We use Adam optimizer for SE-FCN and the momentum is set to 0.9, while SE-PSP and SE-Deeplab use Momentum optimizer and the momentums are both set to 0.9. Due to the limited physical memory on our GPU card, we set the batch-size to 1 during training. Hyper-parameter that controls the weight of regularization in SE-PSP and SE-Deeplab is set to 0.4, which is the same as that in the original PSPNet and Deeplab.

4.4 Zero Shot Scene Parsing (ZSSP)

In this section, we mainly focus on the performance of our SCN on the setting of ZSSP. Here, we do not compare with other ZSL methods on this task, because our concentration is not focused on the design of ZSL algorithms but the application scenario of ZSSP task, and this method can be considered as a baseline for coming researchers.

In our experiment, since there is not a standard public split for seen-unseen classes, we random-repeatedly choose 5 classes as unseen classes, which are not employed during the following training phase. We calculate the averages of mAP and PA of 10 executions, and record the average of them in Tab. 1, and the visual results are shown in Fig. 4. It should be noted that since we have random-repeatedly pick out 5 classes as unseen classes, thus the unseen classes in each raw in Fig. 4 may be different. In addition, the same experiment setting and calculation is exploited in the experiments for GZSSP in the following section.

As shown in Tab. 1, we classify the pixels into 5 unseen categories, and the results obtained by different structures have only a little difference. Among them, SE-Deeplab has the best classification

© The Institution of Engineering and Technology 2015

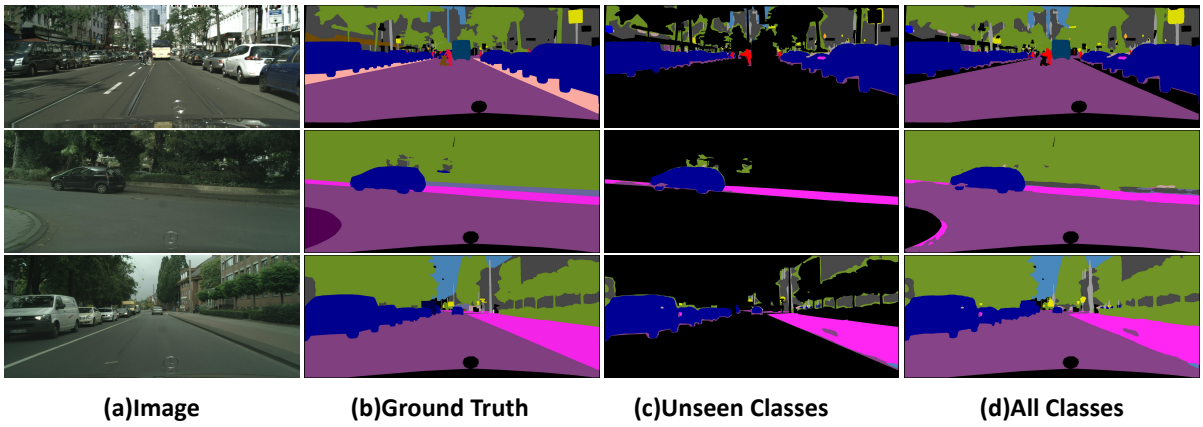

Fig. 5: Some examples of our Generalized Zero Shot Scene Parsing. The first and second row are predictions from SE-PSP, and the last row is from SE-Deeplab.

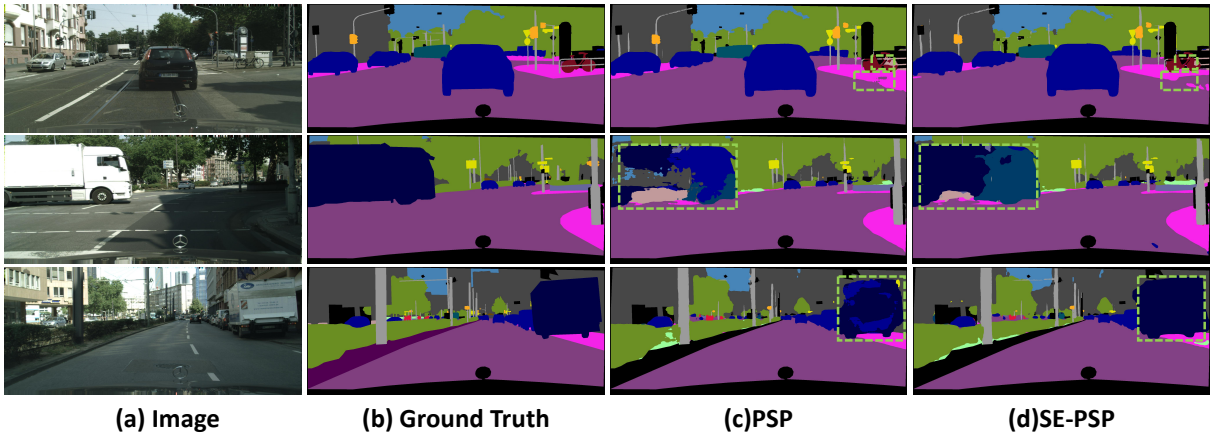

Fig. 6: Visual results of our SE-PSP compared with the original ones.

Table 1 Experiment of SCN for ZSSP with 5 unseen classes. Bold font is for the best result.

Method	PA (%)	mAP (%)
SE-FCN	85.91(±6.2)	53.93(±3.0)
SE-PSP	90.41(±1.8)	59.76 (±1.5)
SE-Deeplab	**91.54**(±1.2)	**62.58**(±0.7)

result, and can reach 62.58% for mAP, and it also has the smallest fluctuation range for the different splits of unseen classes.

Since there is a big unbalance between the number of pixels of each category, which often leads to poor predictions in a category with fewer pixels, and finally causes the degradation of the all class mAP value. However, although mAP in Tab. 1 is not so high, our method can still make a good prediction. As shown in Fig. 4, we can find that although there is some noise in prediction, our method classifies most of the pixels correctly, *e.g.*, *'terrain'* and *'rider'* in the first row, *'rider'* and *'building'* in the second row, and *'car'* and *'sidewalk'* in the last one.

Table 2 Average Result of SCN for GZSSP with 5 unseen classes. Bold font is for the best result.

Method	ts(%)	tr(%)	H
SE-FCN	42.83	54.98	48.15
SE-PSP	42.74	**68.90**	52.76
SE-Deeplab	**45.03**	64.35	**52.98**

Table 3 Pixel Accuracy of each unseen classes under GZSSP setting on SE-Deeplab

Category	Building	Wall	Terrain	Rider	Car
Accuracy(%)	83.30	11.65	17.32	52.03	62.90

4.5 Generalized Zero Shot Scene Parsing (GZSSP)

GZSSP is a more practical task in realistic scenarios. We use the same settings as that in ZSSP, but different searching strategies for classification. In ZSSP, the search range is fixed on 5 unseen classes, while in GZSSP it is relaxed to all 19 classes. Since H is calculated by ts and tr, here we directly compute the final average results of ts (short for acc_{ts}), tr (short for acc_{tr}) and H, which are recorded in Tab. 2, and show the visual category maps in Fig. 5.

From Tab. 2, we can discover that our SCN can obtain good results on both ts and tr, which lead to good result on H. Specifically, SE-DeepLab has the best unseen classes prediction accuracy and SE-PSP predicts seen classes better. SE-PSP and SE-Deeplab have similar overall performance in this task with around 52.8% on H. Since different classes have different numbers of pixels in Cityscapes, especially *'road'* and *'building'* are much more than the others, it will encourage the prediction to be shifted to them during testing, and leads to degradation of mAP, which is the main problem we met in our GZSSP experiment.

In addition, from Fig. 5, we can observe that our SCN can well recognize the unseen class data even though the searching range is expanded to all classes. In the third column, there are some mis-predictions on the boundary of objects, *e.g.*, the border of the *'building'* is misclassified as *'tree'*, which is caused by the contextual information and convolution.

© The Institution of Engineering and Technology 2015

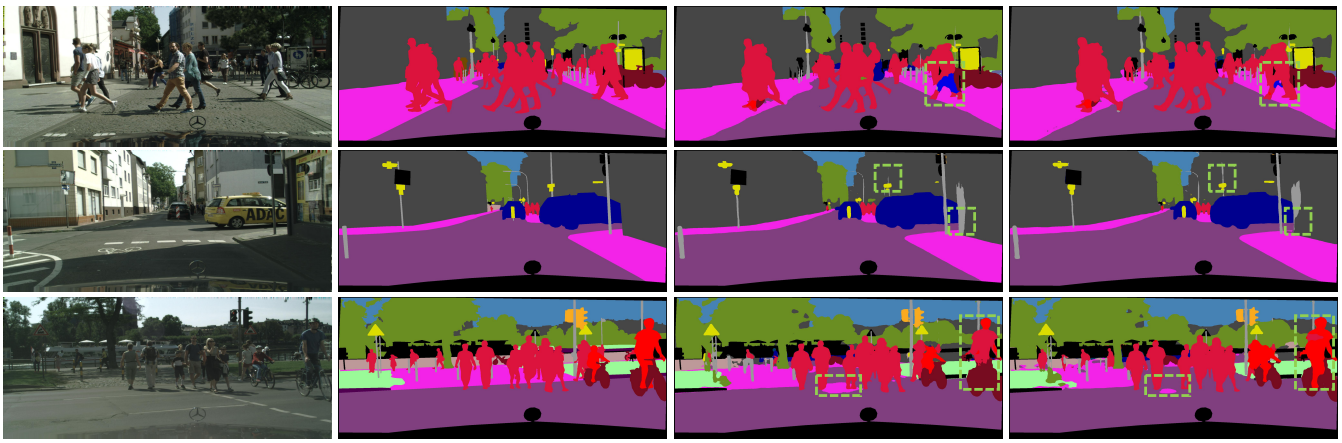

Fig. 7: Visual results of our SE-DeepLab compared with the original ones.

Table 4 Comparison of Class-wise scene parsing between FCN and SE-FCN, PSP and SE-PSP, Deeplab and SE-Deeplab on IoU(The bold fonts in the table indicate the superiority of our results).

Method(%)	Road	Swalk	Build	Wall	Fence	Pole	Tlight	Sign	Veg.	terrain
FCN	96.12	72.00	86.41	26.03	33.11	41.60	36.25	56.21	88.03	49.71
SE-FCN	**97.23**	**78.77**	**89.02**	**38.94**	**42.55**	**51.74**	**51.34**	**65.72**	**90.14**	**56.80**
PSP	97.55	81.67	89.97	47.95	57.31	51.83	60.08	72.53	90.09	58.95
SE-PSP	**97.64**	**82.41**	**91.25**	**52.83**	**59.55**	**54.64**	**66.35**	**74.21**	**91.17**	**61.47**
Deeplab	97.70	82.20	90.38	49.26	58.54	49.72	60.13	71.61	90.64	63.01
SE-Deeplab	**97.88**	**83.57**	**91.12**	**52.58**	**61.28**	**53.99**	**63.51**	**73.74**	**91.23**	**63.68**
Method(%)	**Sky**	**Person**	**Rider**	**Car**	**Truck**	**Bus**	**Train**	**Mbike**	**Bike**	**MIoU**
FCN	91.56	59.38	22.39	87.23	28.05	48.52	31.43	20.91	56.21	54.27
SE-FCN	**93.02**	**69.84**	**38.78**	**91.20**	**40.43**	**52.73**	**38.68**	**38.06**	**66.84**	**62.73**
PSP	90.40	74.71	57.71	92.75	63.04	78.04	59.37	55.21	72.83	71.16
SE-PSP	**91.67**	**77.97**	**60.17**	**93.72**	**72.73**	**81.69**	**64.90**	**61.89**	**74.64**	**74.26**
Deeplab	91.64	75.34	57.08	92.92	62.20	69.98	46.42	58.00	71.94	70.46
SE-Deeplab	**92.45**	**77.48**	**58.65**	**93.52**	**67.54**	**81.47**	**60.29**	**61.97**	**73.50**	**73.66**

Table 5 Comparison of our SCN and original networks for classifying 19 classes. Bold font is for the best result.

Method(%)	Pixel Acc.	mAP	mIoU
FCN	92.28	63.67	54.27
SE-FCN	**94.05**	**71.74**	**62.73**
PSP	94.77	79.76	71.16
SE-PSP	**95.24**	**84.40**	**74.26**
Deeplab	94.89	81.12	70.46
SE-Deeplab	**95.33**	**83.39**	**73.66**

To further illustrate the detailed performance of each category, in Tab. 3, we list the results of SE-Deeplab in one split of unseen classes, which includes *'Building'*, *'Wall'* *'Terrain'* *'Rider'* *'Car'*. We can clearly see that the Pixel Accuracy of *'Building'* is high while the result of *'Wall'* is much lower, which is caused by that *'Building'* and *'Wall'* are belong to the construction and the semantics of *'Building'* and *'Wall'*, directly annotated by word concept, are quite similar, and lead to a bad result on *'Wall'*. Another category that need to be paid close attention is *'Terrain'*, here we also present the accuracy of seen class *'Vegetation'*, which is 96.8%. This phenomenon is caused by the problem of domain shift, which means that trained classifier usually prefer seen classes to unseen classes since the unseen images cannot guide the training process.

4.6 Fully Supervised Scene Parsing

The experimental results are recorded in Tab. 5, from which it can be clearly observed that our SCN outperforms the corresponding original networks. As described above, due to information loss during resizing image, original FCN only get 54.27% for mIoU, but under the same condition our proposed SE-FCN still can get 62.73%, which is a large boost over the original one, and can prove that our network can still preserve the semantic features of each category although under low resolution condition. Concretely, we improves 1.77%, 8.07%, and 8.46% for PA, mAP and mIoU respectively. We also conduct experiments for SE-PSP and SE-Deeplab, and improve 0.47% and 0.32% for PA, 4.64% and 3.1% for mAP, 2.42% and 2.37% for mIoU respectively.

We also calculate the three metrics of each class and illustrate them in Tab. 4 to compare with that of the original methods. As it can be seen, the proposed SCN improves the accuracy on every single category. By combining semantic information, some categories have large difference with others in word concepts are improved a lot, for example *'rider'*, *'truck'*, and *'bus'* have less similarity to other categories in word concept are better classified.

Additionally, we also show some visual results in Fig. 6 and Fig. 7. In the first row of Fig. 6, *'bicycle'* has large semantic difference to *'pole'*, therefore, it can be segmented better. In the second and third row, *'truck'* has more unique semantic information, and our network can predict it more confidently. In the first row of Fig. 7, the leg of *'person'* on the right side is predicted as *'car'* by Deeplab while our method can correctly classify it. In the second row, our result of *'pole'* above the sign on the middle of image is more precise, and the *'building'* on the right side can be predicted better too. In the third row, the *'rider'* on the right of the image is much better classified by our SCN than the original one does.

4.7 Computational Time Analysis

In this section, we test our the computational time of our proposed SCN, and the results are recorded in Tab. 6 and Tab. 7, where all the

© The Institution of Engineering and Technology 2015

Table 6 Computational time of our SCN during training phase, and the measurement unit is second (s).

Settings	(SE-)FCN	(SE-)PSP	(SE-)Deeplab
Traditional	21104	14570	15486
ZSSP&GZSSP	25697	14722	15924
Fully	27988	15377	16424

Table 7 Computational time of testing one figure of our SCN, and the measurement unit is second (s).

Setting	(SE-)FCN	(SE-)PSP	(SE-)Deeplab
Traditional	0.326	0.740	0.802
ZSSP	0.362	0.744	0.809
GZSSP	0.371	0.754	0.814
Fully	0.446	0.767	0.822

experimental results are obtained by utilizing only one single GPU — NVIDIA GTX 1080Ti. To be specific, the values in Tab. 6 are the computational times of total training phase, and those in Tab. 7 are the times of testing one single image, it is noted that we test all three models with the input dimension of 1024×2048. It is known that the smaller image is input the faster computational speed can be obtained. For example, in autonomous navigation, smaller image such as 512×256 is enough in most circumstances, and the computational time can be reduced to its $1/16$, which is sufficient for real-time application. Furthermore, in order to make a comparison with traditional architectures, we also test the training time of original FCN, PSPNet and Deeplab, and the results are shown in the first line of Tab. 6, from which it can be clearly seen that our SCN costs a little more time than the original ones due to the fact that semantic embeddings are combined in by exploiting matrix multiplication, and the dimension of which is 300×19 for each pixel in our network.

5 Conclusion

In this paper, we have proposed a novel and effective network SCN, which combines deep discriminative semantic information from labels with traditional scene parsing architectures to meet the ZSSP setting and the GZSSP tasks, which are challenging tasks of recognizing unseen samples, and the results of our experiments can verify the effectiveness of our proposed method on both ZSL and GZSL settings. Moreover, we also test our method under the fully supervised setting with the utilization of semantic information, our model can recognize scene like a human, and also boost the performance over the original ones that only use visual features. At last, we compute the training and testing time of our model and the results show that our method can be suitable for real-time tasks in some realistic scenarios.

6 Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (61872187), in part by the Medical Research Council (MRC) Innovation Fellowship (MR/S003916/1), and in part by the Natural Science Foundation of Jiangsu Province (BK20160842).

7 References

1 C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.
2 S. H. Khan, M. Bennamoun, F. Sohel, and R. Togneri, "Geometry driven semantic labeling of indoor scenes," in *European Conference on Computer Vision*, 2014, pp. 679–694.
3 J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.
4 H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2881–2890.
5 L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018.
6 X. He, R. S. Zemel, and M. Á. Carreira-Perpiñán, "Multiscale conditional random fields for image labeling," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004.
7 H. Zhang, Y. Long, W. Yang, and L. Shao, "Dual-verification network for zero-shot learning," *Information Sciences*, vol. 470, pp. 43–57, 2019.
8 Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for attribute-based classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 819–826.
9 M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell, "Zero-shot learning with semantic output codes," in *Advances in neural information processing systems*, 2009, pp. 1410–1418.
10 R. Socher, M. Ganjoo, H. Sridhar, O. Bastani, C. D. Manning, and A. Y. Ng, "Zero-shot learning through cross-modal transfer," in *Advances in neural information processing systems*, 2013, pp. 935–943.
11 V. Ferrari and A. Zisserman, "Learning visual attributes," in *NIPS*, 2008, pp. 433–440.
12 H. Zhao, X. Puig, B. Zhou, S. Fidler, and A. Torralba, "Open vocabulary scene parsing," in *The IEEE International Conference on Computer Vision*, Oct 2017, pp. 2002–2010.
13 T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *International Conference on Learning Representation Workshops*, 2013.
14 M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3213–3223.
15 L. Zhou, H. Zhang, Y. Long, L. Shao, and J. Yang, "Depth embedded recurrent predictive parsing network for video scenes," *IEEE Transactions on Intelligent Transportation Systems*, 2019.
16 S. Liu and H. Zhang, "Parallelnet: A depth-guided parallel convolutional network for scene segmentation," in *Pacific Rim International Conference on Artificial Intelligence*, 2018, pp. 588–603.
17 L. Zhou and H. Zhang, "3sp-net: Semantic segmentation network with stereo image pairs for urban scene parsing," in *Pacific Rim International Conference on Artificial Intelligence*, 2018, pp. 503–517.
18 A. Lucchi, Y. Li, X. Boix, K. Smith, and P. Fua, "Are spatial and global constraints really necessary for segmentation?" in *International Conference on Computer Vision*, 2011.
19 K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representation*, 2015.
20 C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.
21 F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *International Conference on Learning Represnetation*, 2016.
22 S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in *International Conference on Computer Vision*, 2015, pp. 1529–1537.
23 Y. Fu, T. M. Hospedales, T. Xiang, Z. Fu, and S. Gong, "Transductive multi-view embedding for zero-shot recognition and annotation," in *European Conference on Computer Vision*, 2014, pp. 584–599.
24 C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 453–465, 2014.
25 Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label embedding for image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 7, pp. 1425–1438, 2016.
26 Z. Akata, F. Perronnin, Z. Harchaoui, and Schmid, "Label embedding for attribute-based classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 819–826.
27 A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov *et al.*, "Devise: A deep visual-semantic embedding model," in *NIPS*, 2013, pp. 2121–2129.
28 M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean, "Zero-shot learning by convex combination of semantic embeddings," in *International Conference on Learning Represnetation*, 2014.
29 Z. Zhang and V. Saligrama, "Zero-shot learning via semantic similarity embedding," in *International Conference on Computer Vision*, 2015, pp. 4166–4174.
30 H. Zhang, Y. Long, L. Liu, and L. Shao, "Adversarial unseen visual feature synthesis for zero-shot learning," *Neurocomputing*, vol. 329, pp. 12–20, 2018.
31 Y. Long, L. Liu, F. Shen, L. Shao, and X. Li, "Transductive zero-shot learning with a self-training dictionary approach," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 10, pp. 2498–2512, 2018.
32 Z. Li, X. Zhang, H. Shen, W. Liang, and Z. He, "A semi-supervised framework for social spammer detection," in *PAKDD*, 2015, pp. 177–188.
33 J. Song, C. Shen, Y. Yang, Y. Liu, and M. Song, "Transductive unbiased embedding for zero-shot learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1024–1033.
34 W.-L. Chao, S. Changpinyo, B. Gong, and F. Sha, "An empirical study and analysis of generalized zero-shot learning for object recognition in the wild," in *European Conference on Computer Vision*, 2016, pp. 52–68.

© The Institution of Engineering and Technology 2015

35 H. Zhang, Y. Long, Y. Guan, and L. Shao, "Triple verification network for generalized zero-shot learning," *IEEE Transactions on Image Processing*, vol. 28, no. 1, pp. 506–517, 2019.
36 Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018.
37 C. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 951–958.
38 S. Changpinyo, W.-L. Chao, B. Gong, and F. Sha, "Synthesized classifiers for zero-shot learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5327–5336.
39 B. Demirel, R. G. Cinbis, and N. Ikizler-Cinbis, "Attributes2classname: A discriminative model for attribute based unsupervised zero-shot learning," in *International Conference on Computer Vision*, 2017.
40 F. Yu, L. Cao, R. S. Feris, J. R. Smith, and S.-F. Chang, "Designing category-level attributes for discriminative visual recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 771–778.
41 Y. Long and L. Shao, "Describing unseen classes by exemplars: Zero-shot learning using grouped simile ensemble," in *IEEE Winter Conference on Applications of Computer Vision*, 2017, pp. 907–915.
42 Z. Zhang and V. Saligrama, "Zero-shot learning via joint latent similarity embedding," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 6034–6042.

© The Institution of Engineering and Technology 2015